HW2 - COPM 329: NLP

Matt Hyatt

Feb 7

Procedure

Preprocessing

I preprocessed the Movie dataset from HW1 first, by converting the entire corpus tp lowercase letters `.lower()` and then used regex `[^/w ]` to remove all punctuation from the text.

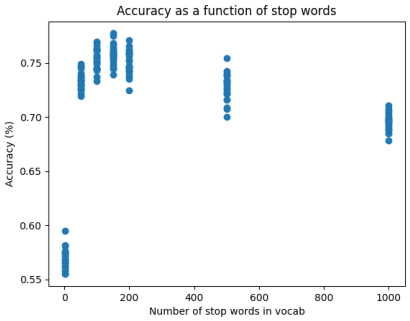
Model

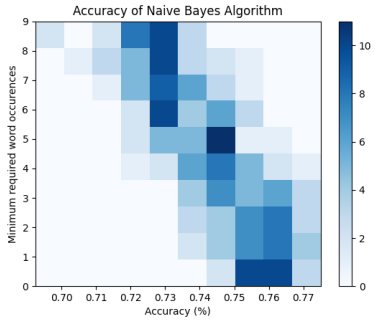
I implemented a Naive Bayes classifier by using dictionaries to count the number of times a word occurred in the training data for each class. I used +1 smoothing to reduce the chance of zero-value logic errors. I used list comprehension to implement an algorithm to calculate P(c|d) for each class c.

Results

Stop Words

I found that my model became more accurate as I began to remove the most common words from the vocabulary. The model's accuracy increased up until about 150 "stop words" were removed from the data. After which the model did not have an optimal amount of information to make decisions on.

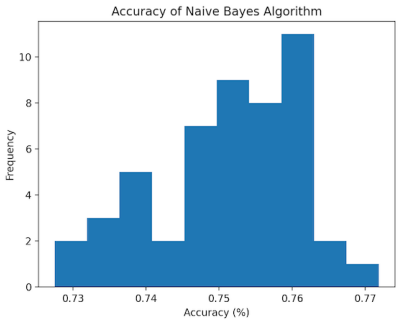


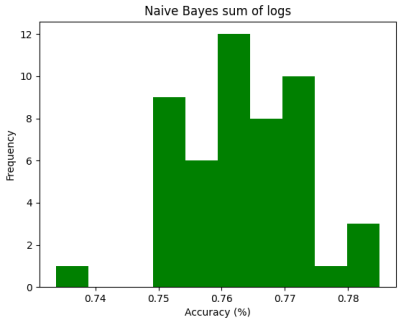


I also removed all words from the vocabulary which had fewer than two occurrences in the training data, although I did not find a significant increase in accuracy after doing so, I could not justify teaching the model to learn from words without any indication that they would reappear.

I was surprised to find that calculating P(c|d) would have any significant differences when calculated in the log space versus when calculated as a product.

After I revised my algorithm to calculate a sum of logs, the mean accuracy of the model increased by a little more than 1%. The complete model recieves an average accuracy of 76%





Class Confidence

Here are some of the examples where my model was most certain:

Document	Class
sort, low, grade, dreck, usually, goes, straight, video, lousy, script, inept, direction, pathetic, acting, poorly, dubbed, dialogue, murky, cinematography, complete, visible, boom	0
dreary, incoherent, self, indulgent, mess, bunch, pompous, hours, pretentious, meaningless	0
script, gem, engaging, intimate, dialogue, realistic, greatly, moving, scope, family, large, grow, attached, lives, full, strength, warmth, vitality	1

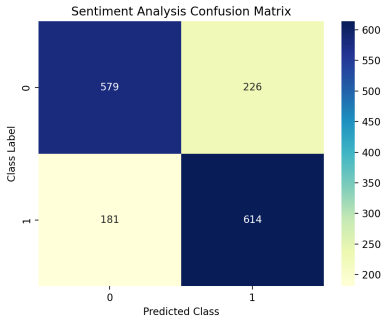
Samples from the negative class contain words like low grade, pathetic acting, dreary, incoherent which have a negative connotation. The positive class contains words like engaging, realistic, moving, attached which have a positive connotation.

Here are some of the examples where my model was least certain:

Document	Class
sham, construct, based, theory, hand, ill, wrought	1
divide, separate, groups, reaching, tissues, begging, mercy	0
wander, predictably, situations, know	0

These samples do not have words with a positive or negative connotation, I would have trouble grading the review as positive or negative by reading them personally. These samples also contain significantly less words than the ones

which are easily classified. Which may give less of an indication.



Important Features

Some of the most important features in the dataset were the following words:

tv, powerful, silly, worst, boring, were, compelling, enjoyable, portrait, documentary, works, moving, world, down, cinema, human, thing, family, entertaining, performance, script, minutes, heart, dull, feels

I took the absolute value of the differences between their respective occurrences in each class. And sorted each word with `list.sort(key=(lambda x: x.val))`